

Introduction to R Markdown

Journalism 303: an introduction to R
markdown

Martin Frigaard

2021-09-21

Rmarkdown

TEXT. CODE. OUTPUT.
(GET IT TOGETHER, PEOPLE.)



Artwork by @allison_horst

Materials



Link to slides:

<https://mjfrigaard.github.io/csuc-data-journalism/slides.html>

Link to exercises:

<https://mjfrigaard.github.io/csuc-data-journalism/lessons-exercises.html>

What is RMarkdown?



Three technologies:

- 1) Markdown is a plain text markup language for capturing *human-readable* prose
- 2) Data manipulation/graphing/statistical language engines for computing *machine-readable* code
- 3) Multiple *output options* for creating PDFs, Word docs, PowerPoints, HTML, etc.

How R Markdown works

rmarkdown works directly with knitr

`rmarkdown` combines YAML, markdown, and R code into a markdown document and passes it to `knitr`



`knitr` uses `pandoc` (a universal document conversion tool) to generate the specified document format





Exercises

We will create an example HTML report using the R Markdown template provided by RStudio

Exercise 1: create a new RMarkdown file

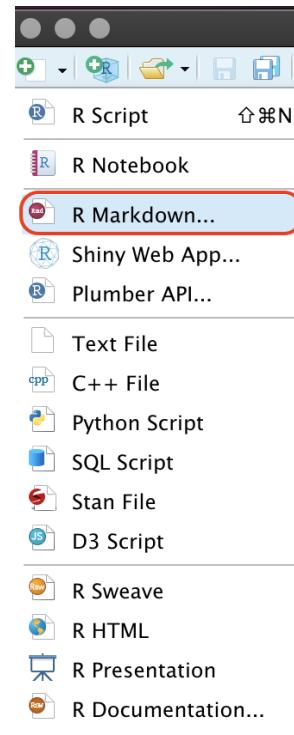


Click on '*File*' >

then '*New File*' >

then '*R Markdown*'

Or use the drop-down menu



Install required packages



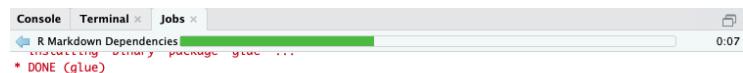
If you're in a fresh RStudio.Cloud session, you *might* be asked to install the required packages for R Markdown, Click **Yes**

Install Required Packages

Creating R Markdown documents requires updated versions of the following packages: Rcpp, base64enc, digest, evaluate, glue, highr, htmltools, jsonlite, knitr, magrittr, markdown, mime, rmarkdown, rprojroot, stringi, stringr, tinytex, xfun, yaml.

Do you want to install these packages now?

You will see RStudio installing the packages in the **Jobs** pane



The screenshot shows the RStudio interface with the 'Jobs' tab selected. A progress bar indicates the download of 'R Markdown Dependencies'. The terminal output shows the installation of the 'highr' package from source, including the URL, content type, and file size. It also shows the download of the 'rlang' package from the package proxy.

```
Console Terminal × Jobs ×
R Markdown Dependencies 0:07
installing binary package 'glue' ...
* DONE (glue)

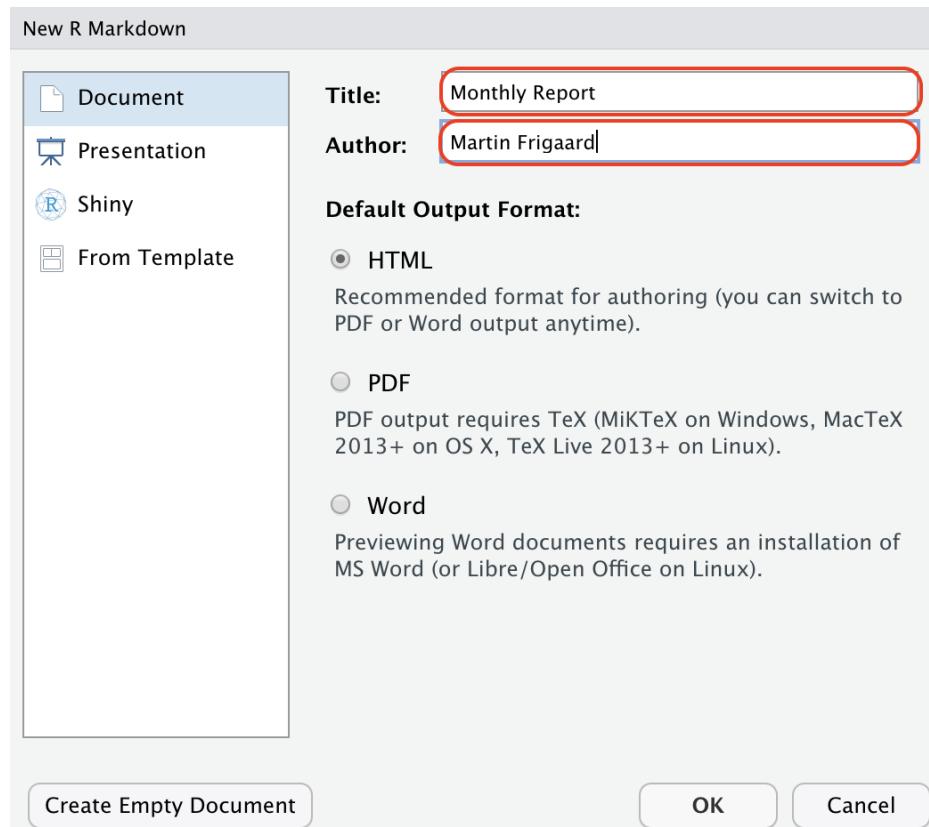
The downloaded source packages are in
'/tmp/RtmpzyUUTC/downloaded_packages'
Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)
trying URL 'http://package-proxy/src/contrib/highr_0.8.tar.gz'
Content type 'application/x-tar' length 40478 bytes (39 KB)
downloaded 39 KB

[7/21] Installing rlang...
* installing *binary* package 'highr' ...
* DONE (highr)

The downloaded source packages are in
'/tmp/RtmpzyUUTC/downloaded_packages'
Installing package into '/home/rstudio-user/R/x86_64-pc-linux-gnu-library/4.0'
(as 'lib' is unspecified)
trying URL 'http://package-proxy/src/contrib/rlang_0.4.8.tar.gz'
```

New R Markdown File

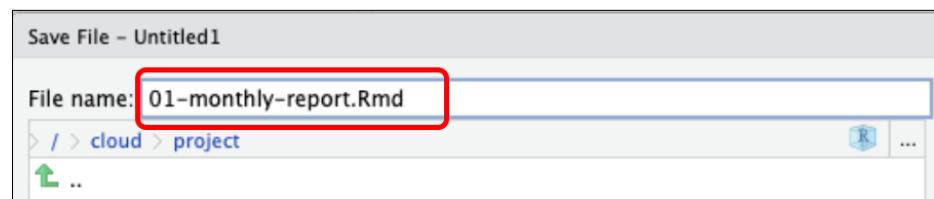
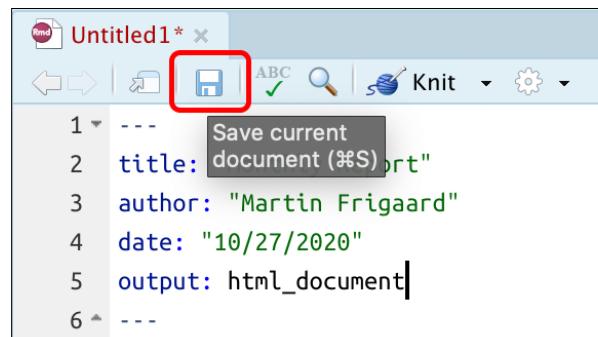
Enter '*Title*' and '*Author*' of your report and click *OK*





Save your .Rmd file

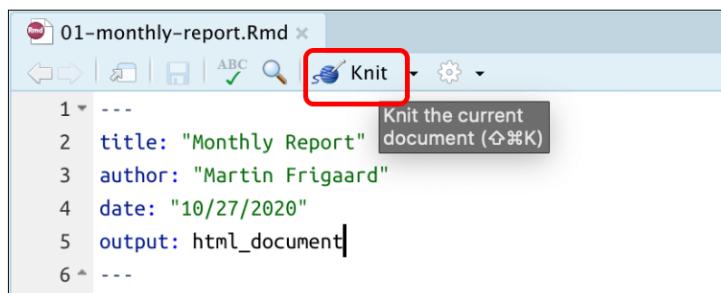
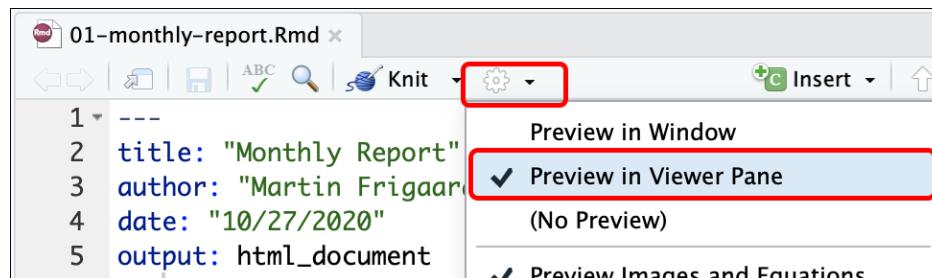
Click on the small floppy disk, enter a name (with .Rmd extension), and save your .Rmd file



Knit your .Rmd file

Click on the small gear, select *Preview in Viewer Pane*

Click on the knit icon (ball of yarn)



Our First R Markdown Report!



Your Workspace / bmrn-04-rmd-intro

File Edit Code View Plots Session Build Debug Profile Tools Help

01-monthly-report.Rmd x

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output: html_document  
6 ---  
7  
8 ```{r setup, include=FALSE}  
9 knitr::opts_chunk$set(echo = TRUE)  
10```  
11  
12 ## R Markdown  
13  
14 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.  
15  
16 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:  
17  
18 ```{r cars}  
19 summary(cars)  
20```  
21  
22 ## Including Plots  
23  
24 You can also embed plots, for example:  
25  
26 ```{r pressure, echo=FALSE}  
27 plot(pressure)  
28```  
29  
30 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.  
31
```

Insert Run Knit

Environment History Connections Tutorial

Files Plots Packages Help Viewer

R 4.0.3

Monthly Report

Martin Frigaard
10/27/2020

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##   speed      dist  
## Min.   : 4.0   Min.   :  2.00  
## 1st Qu.:12.0   1st Qu.: 26.00  
## Median :15.0   Median : 36.00  
## Mean    :15.4   Mean   : 42.98  
## 3rd Qu.:19.0   3rd Qu.: 56.00  
## Max.   :25.0   Max.   :120.00
```

Including Plots

You can also embed plots, for example:

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.



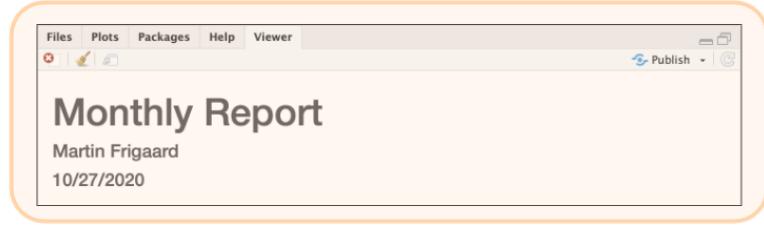
How R Markdown Works (under the hood)

R Markdown is made up of three elements

YAML header = metadata

A screenshot of the RStudio interface showing the YAML header of an R Markdown file named "01-monthly-report.Rmd". The code is:

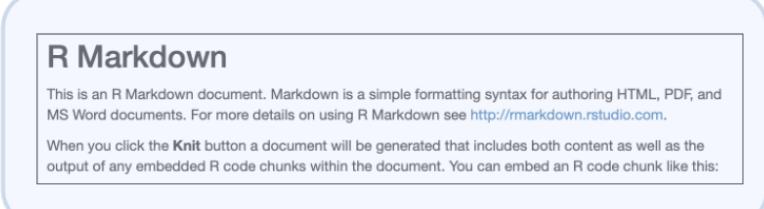
```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output: html_document  
6 ---  
7
```



Markdown = prose

A screenshot of the RStudio interface showing an R Markdown document. The text includes:

```
12 ## R Markdown  
13  
14 This is an R Markdown document. Markdown is a simple formatting syntax for  
authoring HTML, PDF, and MS Word documents. For more details on using R Markdown  
see <http://rmarkdown.rstudio.com>.  
15  
16 When you click the **Knit** button a document will be generated that includes both  
content as well as the output of any embedded R code chunks within the document.  
You can embed an R code chunk like this:
```



Code chunks = R code

A screenshot of the RStudio interface showing an R Markdown document with an R code chunk:

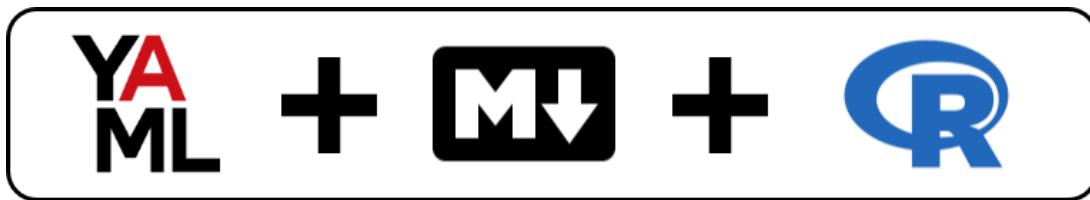
```
18 ````{r cars}  
19 summary(cars)  
20 ````
```

A screenshot of the RStudio interface showing the output of the R code chunk. The output is the summary statistics for the "cars" dataset:

```
summary(cars)
```

	speed	dist
## Min.	4.0	2.00
## 1st Qu.	12.0	1st Qu.: 26.00
## Median	15.0	Median : 36.00
## Mean	15.4	Mean : 42.98
## 3rd Qu.	19.0	3rd Qu.: 56.00
## Max.	25.0	Max. : 120.00

Rmarkdown combines metadata, markdown, and R code



The result is a file framework for creating reproducible reports using YAML, Markdown, and computer code

- .yaml = Metadata
- .md = Prose
- .R = Code

R Markdown: YAML



- **.yaml** = Metadata

- ~~• .md = Prose~~

- ~~• .R = Code~~

YAML is a human friendly data serialization standard for all programming languages.

YAML stands for '*YAML Ain't Markup Language*'
(funny, huh?)

R Markdown: YAML



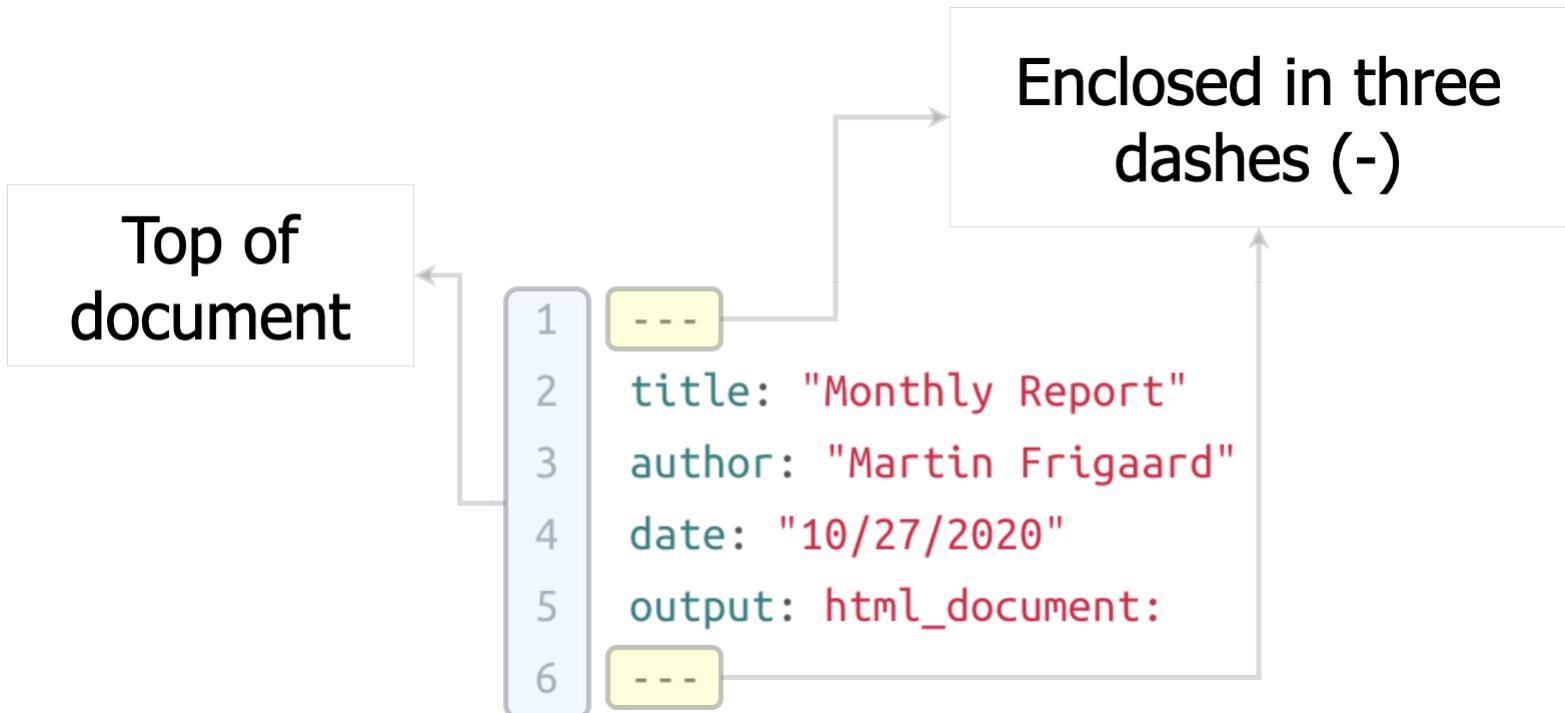
YAML contains the information about the document we're going to create

```
---
```

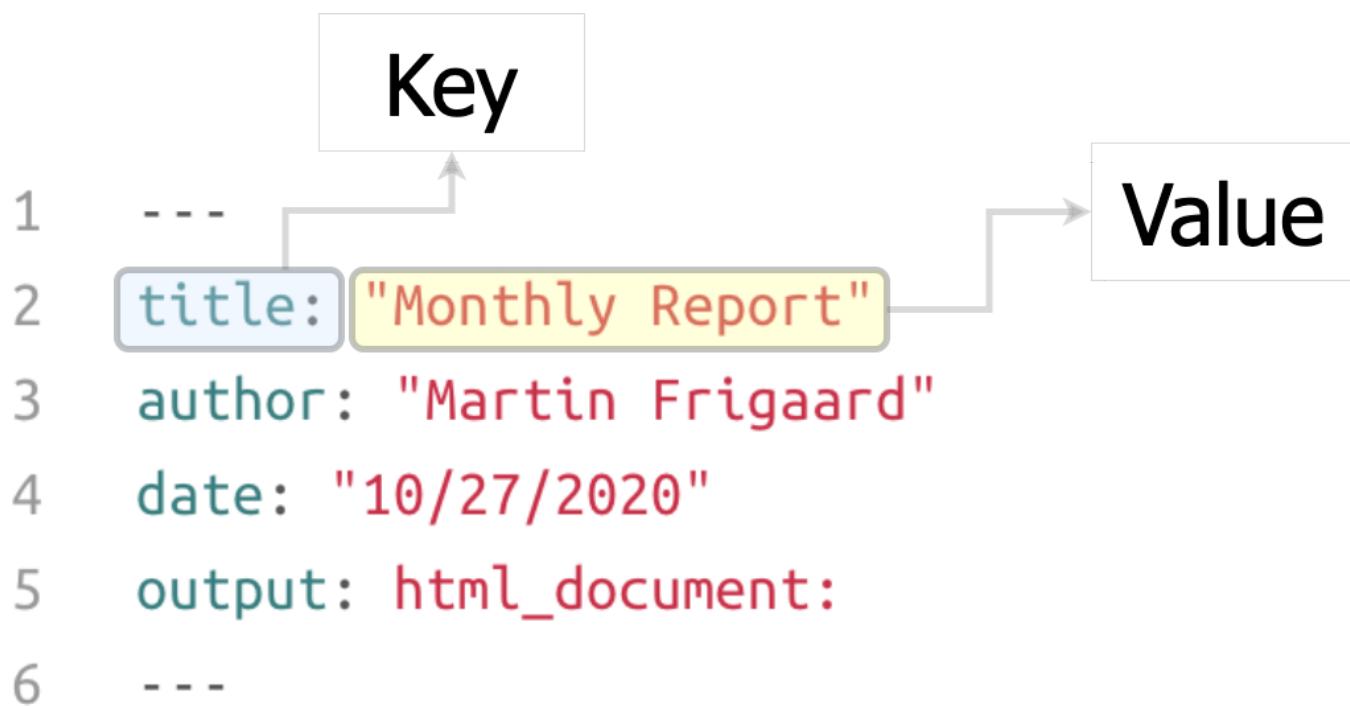
```
title: "Monthly Report"
author: "Martin Frigaard"
date: "10/27/2020"
output: html_document
```

```
---
```

R Markdown: YAML format



R Markdown: YAML format



R Markdown: YAML



There are many YAML arguments and options

Indentation matters in YAML!!

| Check out the [YAML Fieldguide](#) for a comprehensive list

Example YAML output options

Table of contents:

toc: logical (true or false)

toc_float: logical (true or false)

toc_depth: set numerically 0 - 6



Exercise 2: create a floating table of contents

Change the **output** in the YAML header to the following:

```
output:  
  html_document:  
    toc: yes  
    toc_float: true
```

Knit the document again



YAML output options: table of contents

Floating table of contents (rendered)

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: true
```

The diagram illustrates the connection between the YAML configuration and the resulting rendered document. An orange arrow points from the 'toc_float: true' line in the YAML code to the floating table of contents in the rendered output.

R Markdown
Including Plots

Monthly Report

Martin Frigaard

10/27/2020

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:



Exercise 3: text highlighting and themes

Add the following two options to your YAML header

```
output:  
  html_document:  
    toc: yes  
    toc_float: yes  
    highlight: zenburn  
    theme: united
```

Knit the document again



YAML: text highlighting and theme options

Text highlighting and theme options (rendered)

The image shows a comparison between the YAML configuration file and its rendered output in RStudio.

YAML Configuration:

```
1 - ---  
2   title: "Monthly Report"  
3   author: "Martin Frigaard"  
4   date: "10/27/2020"  
5   output:  
6     html_document:  
7       toc: yes  
8       toc_float: yes  
9       highlight: zenburn  
10      theme: united  
11 - ---
```

RStudio Viewer Output:

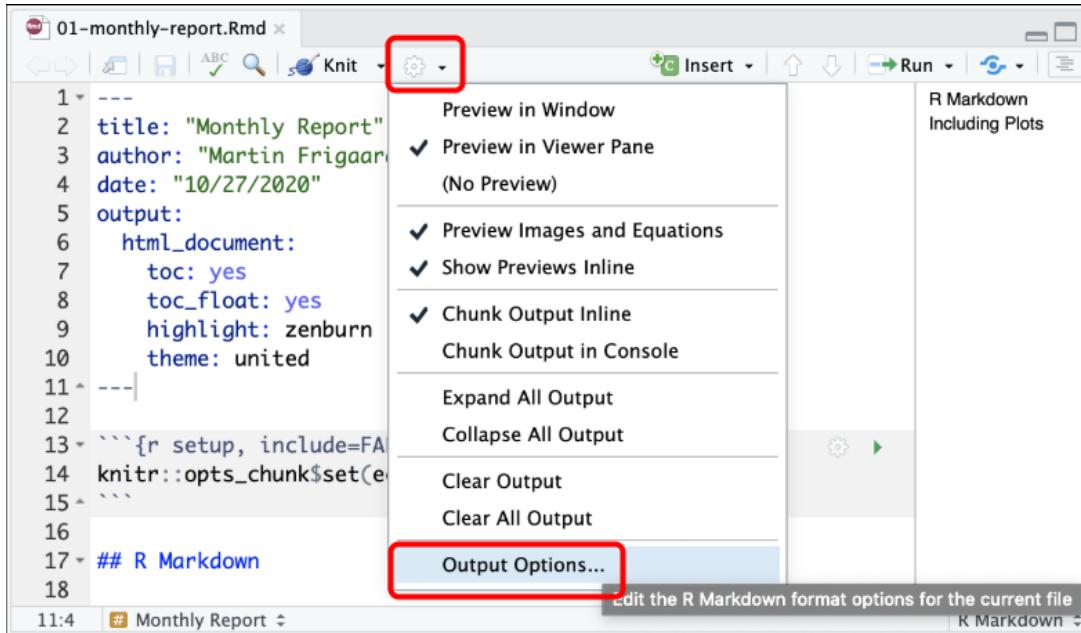
The RStudio interface shows the rendered document. The title bar says "Monthly Report". The header includes "Martin Frigaard" and the date "10/27/2020". A section titled "R Markdown" explains what it is. Below it, a code chunk titled "summary(cars)" is shown, with its output being a table of car speeds and distances.

	speed	dist
## Min.	4.0	Min. : 2.00
## 1st Qu.	12.0	1st Qu.: 26.00
## Median	15.0	Median : 36.00
## Mean	15.4	Mean : 42.98
## 3rd Qu.	19.0	3rd Qu.: 56.00
## Max.	25.0	Max. : 120.00



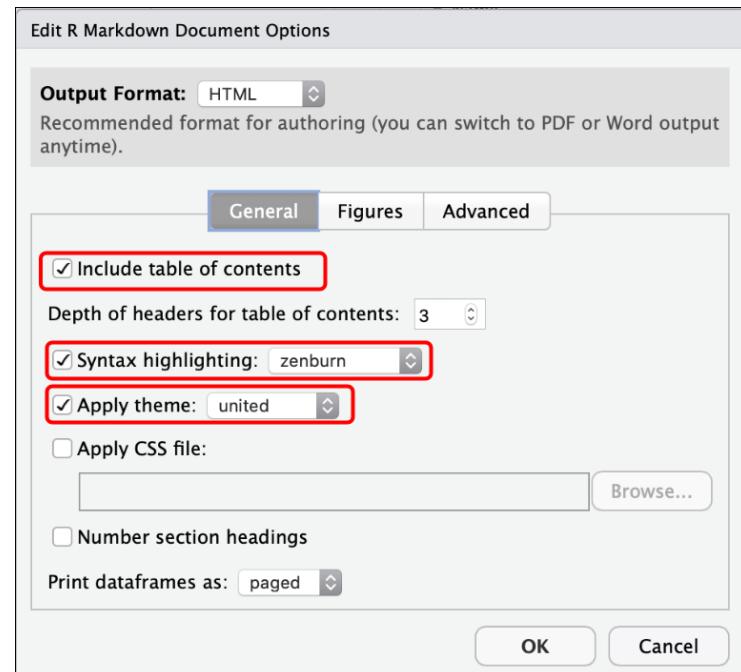
More YAML options

You can change the YAML contents using the settings (small gear)



Edit R Markdown Document Options

This window gives us the ability to manually change some of the YAML settings (but not all of them!)



YAML Parameters

YAML parameters can be referred to throughout the document

Create params in YAML header

```
params:  
  param1: x  
  param2: y  
  data: df
```

Refer to params in .Rmd document

```
params$param1  
params$param2  
params$data
```



Exercise 4: Using YAML parameters

Add the following `params` option in the YAML header

```
params:  
  small_pressure: !r head(pressure)
```

Add this code to the end of the document

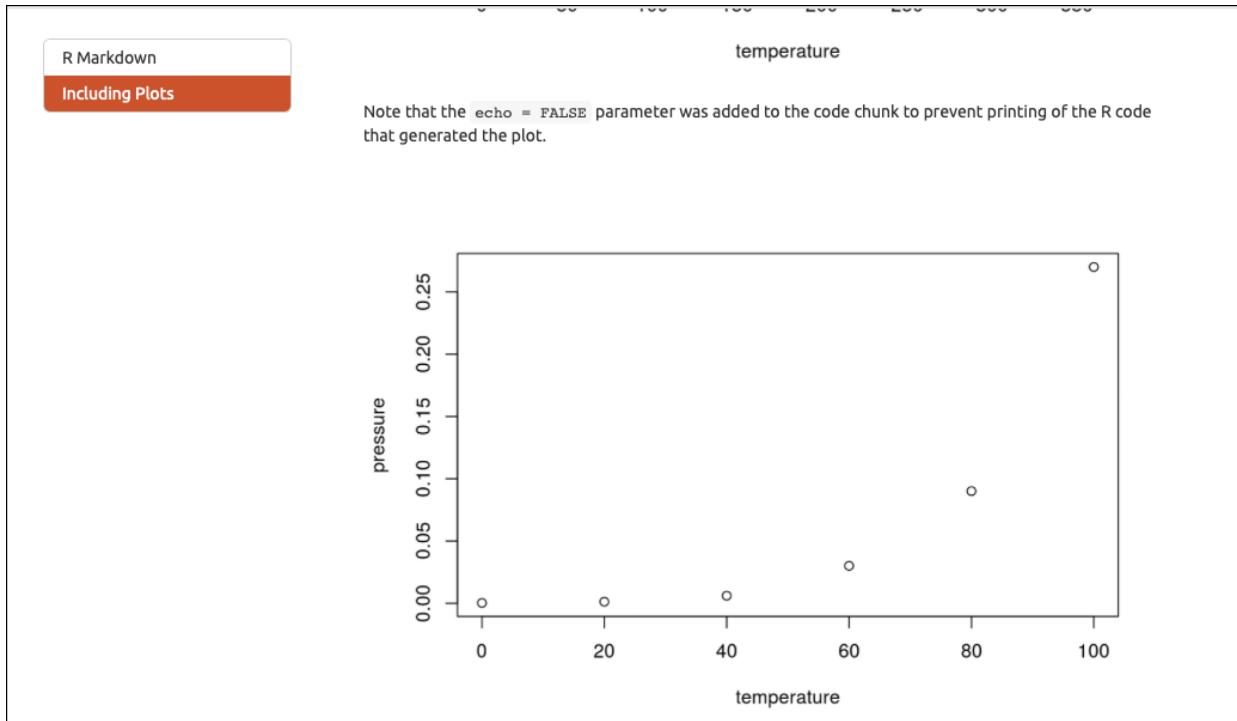
```
```{r small_pressure, echo=FALSE}  
plot(params$small_pressure)
```
```

Knit the document again



See our new plot with the params

We can see the new plot with the reduced sample size



YAML output formats



| Function | Outputs |
|--|---------------------|
| <code>html_document()</code> | HTML document |
| <code>pdf_document()</code> | PDF document |
| <code>word_document()</code> | Word document |
| <code>odt_document()</code> | ODT document |
| <code>rtf_document()</code> | RTF document |
| <code>md_document()</code> | Markdown document |
| <code>slidy_presentation()</code> | Slidy Slides (HTML) |
| <code>beamer_presentation()</code> | Beamer Slides (PDF) |
| <code>ioslides_presentation()</code> | ioslides (HTML) |
| <code>powerpoint_presentation()</code> | PowerPoint (pptx) |

R Markdown

~~.yaml = Metadata~~



.md = Prose

~~.R = Code~~

Basic Markdown Syntax



Italics & Bold

```
*italic*    **bold**  
_italic_    __bold__
```

italic **bold**
italic **bold**

Basic Markdown Syntax



Headers

```
# Header 1  
## Header 2  
### Header 3
```

Header 1

Header 2

Header 3

Basic Markdown Syntax



Bullets & Numeric Lists

```
* Item 1  
* Item 2  
    + Item 2a  
    + Item 2b
```

```
1. Item 1  
2. Item 2
```

- Item 1
- Item 2
 - Item 2a
 - Item 2b

1. Item 1
2. Item 2

Basic Markdown Syntax



Hyperlinks

```
https://www.biamarin.com/
```

```
[linked phrase] (https://www.biamarin.com/)
```

becomes...

linked phrase

Basic Markdown Syntax



Images

```
![] (https://www.r-project.org/logo/Rlogo.png)
```

```
![optional caption] (https://www.r-project.org/logo/Rlogo.png)
```

becomes...



Basic Markdown Syntax



Math Equations

```
$equation$
```

```
$$ equation $$
```

equation

equation

Basic Markdown Syntax



Super scripts & Strike-through

```
superscript^2^  
~~strikethrough~~
```

superscript^2^

~~strikethrough~~

Exercise 5: Markdown Formatting

Delete the top portion of the markdown in `01-monthly-report.Rmd`.

Delete {

```
1 --
2 title: "Monthly Report"
3 author: "Martin Frigaard"
4 date: "10/27/2020"
5 output:
6   html_document:
7     toc: yes
8     toc_float: yes
9     highlight: zenburn
10    theme: united
11 params:
12   small_pressure: !r head(pressure)
13 ---
14
15 ``{r setup, include=FALSE}
16 knitr::opts_chunk$set(echo = TRUE)
17 ```
18
19 ## R Markdown
20
21 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.
22
23 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```



Exercise 5: Markdown Formatting

Add the text below to your report

This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.

Include the following formatting:

1. make `monthly report` italic
2. include this hyperlink for `Rmarkdown`: <https://rmarkdown.rstudio.com/>
3. format `code` as code.

Knit the document when you're finished



Exercise 5: Markdown Formatting (rendered)

The screenshot shows the RStudio interface with an R Markdown file named "01-monthly-report.Rmd" on the left and its rendered output on the right.

Code Editor (Left):

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "10/27/2020"  
5 output:  
6   html_document:  
7     toc: yes  
8     toc_float: yes  
9     highlight: zenburn  
10    theme: united  
11 params:  
12   small_pressure: !r head(pressure)  
13 ---  
14  
15 ````{r setup, include=FALSE}  
16 knitr::opts_chunk$set(echo = TRUE)  
17 ````  
18  
19 ## R Markdown  
20  
21 This is a *monthly report* generated with  
[RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool  
for combining text and `code`.  
22  
23 ````{r cars}  
24 summary(cars)  
25 ````
```

R Markdown Output (Right):

Header:

Monthly Report
Martin Frigaard
10/27/2020

Section:

R Markdown

This is a *monthly report* generated with **RMarkdown**, a literate programming tool for combining text and `code`.

Code Block:

```
summary(cars)
```

Data Output:

| | speed | dist |
|------------|-------|----------------|
| ## Min. | 4.0 | Min. : 2.00 |
| ## 1st Qu. | 12.0 | 1st Qu.: 26.00 |
| ## Median | 15.0 | Median : 36.00 |
| ## Mean | 15.4 | Mean : 42.98 |
| ## 3rd Qu. | 19.0 | 3rd Qu.: 56.00 |
| ## Max. | 25.0 | Max. :120.00 |



Exercise 6: Tabssets



Remove the `toc` and `toc_float` options from your YAML header

```
output:  
  html_document:  
    highlight: zenburn  
    theme: united  
params:  
  small_pressure: !r head(pressure)
```

Exercise 6: Tabssets



Make the following changes to the R Markdown header sections

```
18  
19 - ## R Markdown {.tabset}  
20  
21 This is a **monthly report** generated with  
[RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool  
for combining text and `code`.  
22  
23 - ### Summary  
24  
25 - ```{r cars}  
26 summary(cars)  
27 ^ ````  
28  
29 - ### Including Plots  
30  
31 You can also embed plots, for example:  
32  
33 - ```{r pressure, echo=FALSE}  
34 plot(pressure)  
35 ^ ````  
36  
37 Note that the `echo = FALSE` parameter was added to the code chunk to  
prevent printing of the R code that generated the plot.  
38  
39 - ```{r small_pressure, echo=FALSE}  
40 plot(params$small_pressure)  
41 ^ ````
```

Include tabset in curly brackets

Add 'Summary' level 3 header

Change 'Including Plots' to level 3 header

Knit the document again

Exercise 6: Tabssets (rendered)



Tab 1

A screenshot of the RStudio interface. On the left, the code editor shows an R Markdown file named "01-monthly-report.Rmd". The code includes YAML front matter, a code chunk for "small_pressure", and a code chunk for "cars" with a summary table. On the right, the "Viewer" panel displays the rendered "Monthly Report". The title is "Monthly Report" by "Martin Frigaard" on "11/5/2020". Below the title is the section "R Markdown" with a subtitle "This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.". There are two tabs: "Summary" (selected) and "Including Plots". A code block shows the "summary(cars)" command, and the resulting table is displayed below it.

```
1 ---  
2 title: "Monthly Report"  
3 author: "Martin Frigaard"  
4 date: "11/5/2020"  
5 output:  
6   html_document:  
7     highlight: zenburn  
8     theme: united  
9 params:  
10   small_pressure: !r head(pressure)  
11 - ---  
12  
13 ````{r setup, include=FALSE}  
14 knitr::opts_chunk$set(echo = TRUE)  
15 ````  
16  
17 ## R Markdown {.tabset}  
18  
19 This is a **monthly report** generated with [RMarkdown](https://rmarkdown.rstudio.com/), a  
literate programming tool for combining text and `code`.  
20  
21 ### Summary  
22  
23 ````{r cars}  
24 summary(cars)  
25 ````  
26  
27 ### Including Plots  
28  
29 You can also embed plots, for example:  
30  
31 ````{r pressure, echo=FALSE}  
32 plot(pressure)  
33 ````  
34  
35 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R  
code that generated the plot.  
36  
37 ````{r small_pressure, echo=FALSE}  
38 plot(params$small_pressure)  
39 ````  
40  
41
```

Monthly Report

Martin Frigaard

11/5/2020

R Markdown

This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.

Summary Including Plots

summary(cars)

| | speed | dist |
|------------|-------|----------------|
| ## Min. | 4.0 | 2.00 |
| ## 1st Qu. | 12.0 | 1st Qu.: 26.00 |
| ## Median | 15.0 | Median : 36.00 |
| ## Mean | 15.4 | Mean : 42.98 |
| ## 3rd Qu. | 19.0 | 3rd Qu.: 56.00 |
| ## Max. | 25.0 | Max. :120.00 |

Exercise 6: Tabs (rendered)



Tab 2

A screenshot of the RStudio interface. On the left, the code editor shows an R Markdown file named "01-monthly-report.Rmd". The code includes YAML front matter, a section for "R Markdown (.tabset)", and a section for "Including Plots". It also contains R code chunks for `summary(cars)` and `plot(pressure)`, with a note about the `echo = FALSE` parameter. On the right, the "Viewer" panel displays the rendered "Monthly Report". The title is "Monthly Report" by Martin Frigaard, dated 11/5/2020. Below the title, there are two tabs: "Summary" (selected) and "Including Plots". A note states: "This is a monthly report generated with RMarkdown (https://rmarkdown.rstudio.com/), a literate programming tool for combining text and code.". Under the "Summary" tab, there is a placeholder text: "You can also embed plots, for example:". Below the viewer, a scatter plot titled "pressure" vs "temperature" is shown, displaying a positive correlation between the two variables.

01-monthly-report.Rmd

```
1 - ---
2   title: "Monthly Report"
3   author: "Martin Frigaard"
4   date: "11/5/2020"
5   output:
6     html_document:
7       highlight: zenburn
8       theme: united
9   params:
10    small_pressure: !r head(pressure)
11  ---
12
13  ```{r setup, include=FALSE}
14  knitr::opts_chunk$set(echo = TRUE)
15  ```
16
17  ## R Markdown {.tabset}
18
19 This is a **monthly report** generated with [RMarkdown](https://rmarkdown.rstudio.com/), a literate programming tool for combining text and code.
20
21  ### Summary
22
23  ```{r cars}
24  summary(cars)
25  ```
26
27  ### Including Plots
28
29 You can also embed plots, for example:
30
31  ```{r pressure, echo=FALSE}
32  plot(pressure)
33  ```
34
35 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.
36
37  ```{r small_pressure, echo=FALSE}
38  plot(params$small_pressure)
39  ```
40
41
```

Environment History Connections Tutorial

Files Plots Packages Help Viewer

Monthly Report

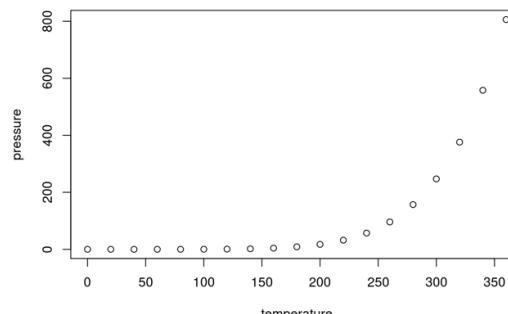
Martin Frigaard
11/5/2020

R Markdown

This is a monthly report generated with RMarkdown, a literate programming tool for combining text and code.

[Summary](#) [Including Plots](#)

You can also embed plots, for example:



A scatter plot showing the relationship between temperature (x-axis, ranging from 0 to 350) and pressure (y-axis, ranging from 0 to 800). The data points show a clear positive correlation, starting near (0, 0) and ending near (350, 800).

| temperature | pressure |
|-------------|----------|
| 0 | 20 |
| 50 | 20 |
| 100 | 20 |
| 150 | 20 |
| 200 | 20 |
| 250 | 50 |
| 300 | 200 |
| 350 | 800 |

R Markdown



~~.yaml = Metadata~~

~~.md = Prose~~

.R = Code

Code chunks (**setup**)



The first bit of R code in our RMarkdown file is the **setup** chunk

```
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```



Chunks named '**setup**' are special because they can set global options

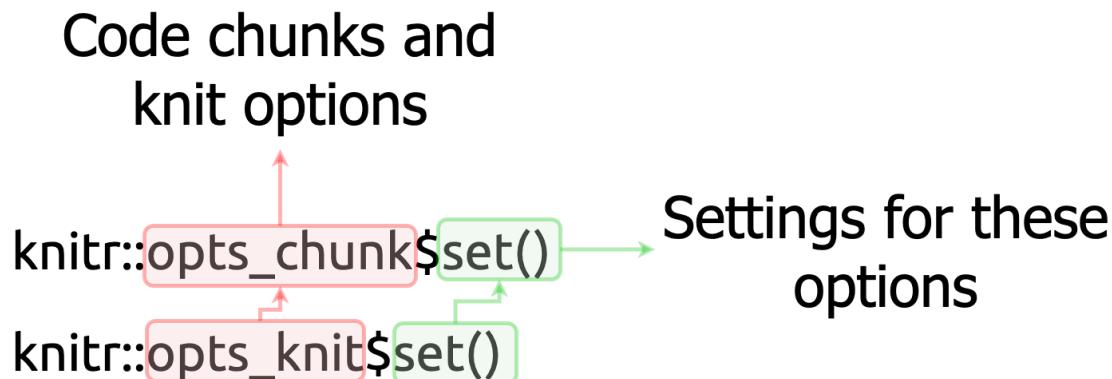
'include=FALSE' means this code is run, but not displayed

Code chunks (`setup`)



R Markdown document options come from the `knitr` package

We can access both the with syntax below:



Code chunks (`setup`)



The `echo=TRUE` option controls whether we want to display the code in the code chunk

Other common options regarding code are `eval`, `tidy`, `error`, `message`, and `warning`

Advanced options can control language engines (`engine`), caching (`cache`, `dependson`), and plot animations (`fig.show`)

Code chunks (setup)



Many options for code chunks

| Option | Document Effect |
|----------------------|---|
| <code>include</code> | run code, but do/don't print code and results |
| <code>eval</code> | do/don't evaluate the code |
| <code>echo</code> | run code, but don't print code |
| <code>message</code> | do/don't print messages (e.g. from functions) |
| <code>warning</code> | do/don't print warnings |

Code chunks



```
1   ```{r pressure, echo=FALSE}
2   plot(pressure)
3   ````
```

Code chunk fences



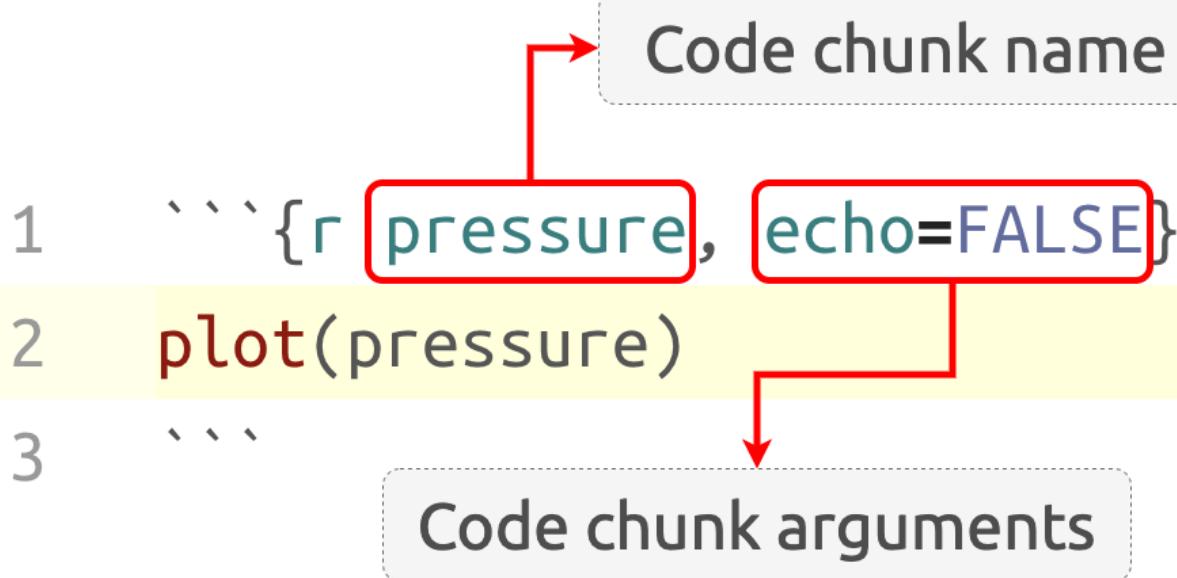
Code chunk fences

```
1  ````{r pressure, echo=FALSE}
```

```
2  plot(pressure)
```

```
3  ````
```

Code chunk names and arguments

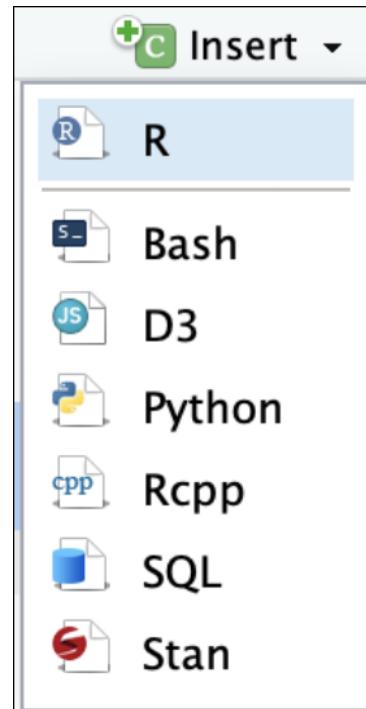


See the knitr web page for complete list of options

Inserting code chunks



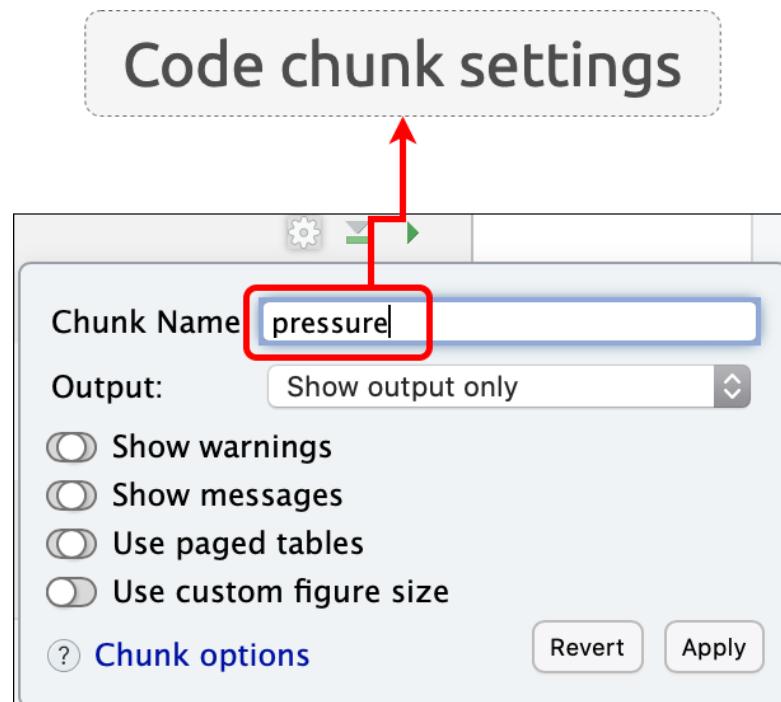
Use keyboard shortcuts CMD/CTRL + I or
ALT/OPTION + I



Edit code chunk options



You can edit code chunk options using the icon (small gear)



Code Chunk Engines



More and more code engines all the time

```
names(knitr::knit_engines$get())
```

```
## [1] "awk"          "bash"         "coffee"        "gawk"         "groovy"       "haskell"
## [7] "lein"         "mysql"        "node"          "octave"       "perl"         "psql"
## [13] "Rscript"      "ruby"         "sas"           "scala"        "sed"          "sh"
## [19] "stata"        "zsh"          "highlight"    "Rcpp"         "tikz"         "dot"
## [25] "c"            "cc"           "fortran"      "fortran95"   "asy"          "cat"
## [31] "asis"         "stan"         "block"         "block2"       "js"           "css"
## [37] "sql"          "go"           "python"       "julia"        "sass"         "scss"
## [43] "R"            "bslib"        "targets"
```

Even SAS!

Exercise 7: code chunks (kable)

Create a new heading and code chunk

Create a new **Tables** level three header under the **Summary** heading,

```
### Tables
```

Insert the following code chunk under **Tables** (*manually, with the keyboard short-cut, or use the "Insert" button*)

```
```{r kable}
knitr::kable(params$small_pressure)
```
```

Knit the document



Exercise 7: code chunks (kable rendered)



We can see the `small_pressure` parameter from the YAML has been rendered in the new **Tables** tab. **kable** tables are great for presenting small, summary tables.

The screenshot shows the R Markdown interface. On the left, a code editor window displays the following R code:## Tables
```{r kable}
knitr::kable(params\$small\_pressure)
```
A red box highlights the first two lines of code. A red arrow points from this box to the 'Tables' tab in the navigation bar on the right. The 'Tables' tab is selected, and the rendered output is displayed below it. The output shows the following table:| | temperature | pressure |
| --- | --- | --- |
| 0 | 0.0002 |
| 20 | 0.0012 |
| 40 | 0.0060 |
| 60 | 0.0300 |
| 80 | 0.0900 |
| 100 | 0.2700 |

Read more about **kable** table options [here](#)

Exercise 8: code chunks (paged)

We are going to repeat the process above, but with a larger table (`mtcars`)

Insert the following code chunk above the `knitr::kable()` output:

```
### Tables
```

```
```{r paged}
rmarkdown::paged_table(mtcars)
```
```



```
```{r kable}
knitr::kable(params$small_pressure)
```
```



Knit the document

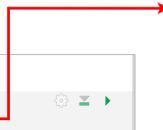


Exercise 8: code chunk (paged rendered)

Paged tables are great for larger datasets

```
### Tables
```{r paged}
rmarkdown::paged_table(mtcars)
```
```{r kable}
knitr::kable(params$small_pressure)
```

```



R Markdown

This is a monthly report generated with **RMarkdown**, a literate programming tool for combining text and code .

Summary Tables Including Plots

```
rmarkdown::paged_table(mtcars)
```

| | mpg | cyl | disp | hp | drat | wt | qsec | vs | am |
|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | <dbl> |
| Mazda RX4 | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.620 | 16.46 | 0 | 1 |
| Mazda RX4 Wag | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.875 | 17.02 | 0 | 1 |
| Datsun 710 | 22.8 | 4 | 108.0 | 93 | 3.85 | 2.320 | 18.61 | 1 | 1 |
| Hornet 4 Drive | 21.4 | 6 | 258.0 | 110 | 3.08 | 3.215 | 19.44 | 1 | 0 |
| Hornet Sportabout | 18.7 | 8 | 360.0 | 175 | 3.15 | 3.440 | 17.02 | 0 | 0 |
| Valiant | 18.1 | 6 | 225.0 | 105 | 2.76 | 3.460 | 20.22 | 1 | 0 |
| Duster 360 | 14.3 | 8 | 360.0 | 245 | 3.21 | 3.570 | 15.84 | 0 | 0 |
| Merc 240D | 24.4 | 4 | 146.7 | 62 | 3.69 | 3.190 | 20.00 | 1 | 0 |
| Merc 230 | 22.8 | 4 | 140.8 | 95 | 3.92 | 3.150 | 22.90 | 1 | 0 |
| Merc 280 | 19.2 | 6 | 167.6 | 123 | 3.92 | 3.440 | 18.30 | 1 | 0 |

1-10 of 32 rows | 1-10 of 12 columns Previous 1 2 3 4 Next

The rmarkdown logo is a red hexagon containing a white icon of a quill pen with an 'R' on it.

61 / 68

Exercise 8: paged tables



R Markdown

This is a **monthly report** generated with **RMarkdown**, a literate programming tool for combining text and `code`.

Summary

Tables

Including Plots

```
rmarkdown::paged_table(mtcars)
```

| | mpg
<dbl> | cyl
<dbl> | disp
<dbl> | hp
<dbl> | drat
<dbl> | wt
<dbl> | qsec
<dbl> | vs
<dbl> | am
<dbl> | ► |
|-------------------|--------------|--------------|---------------|-------------|---------------|-------------|---------------|-------------|-------------|---|
| Mazda RX4 | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.620 | 16.46 | 0 | 1 | |
| Mazda RX4 Wag | 21.0 | 6 | 160.0 | 110 | 3.90 | 2.875 | 17.02 | 0 | 1 | |
| Datsun 710 | 22.8 | 4 | 108.0 | 93 | 3.85 | 2.320 | 18.61 | 1 | 1 | |
| Hornet 4 Drive | 21.4 | 6 | 258.0 | 110 | 3.08 | 3.215 | 19.44 | 1 | 0 | |
| Hornet Sportabout | 18.7 | 8 | 360.0 | 175 | 3.15 | 3.440 | 17.02 | 0 | 0 | |
| Valiant | 18.1 | 6 | 225.0 | 105 | 2.76 | 3.460 | 20.22 | 1 | 0 | |
| Duster 360 | 14.3 | 8 | 360.0 | 245 | 3.21 | 3.570 | 15.84 | 0 | 0 | |
| Merc 240D | 24.4 | 4 | 146.7 | 62 | 3.69 | 3.190 | 20.00 | 1 | 0 | |
| Merc 230 | 22.8 | 4 | 140.8 | 95 | 3.92 | 3.150 | 22.90 | 1 | 0 | |
| Merc 280 | 19.2 | 6 | 167.6 | 123 | 3.92 | 3.440 | 18.30 | 1 | 0 | |

1-10 of 32 rows | 1-10 of 12 columns

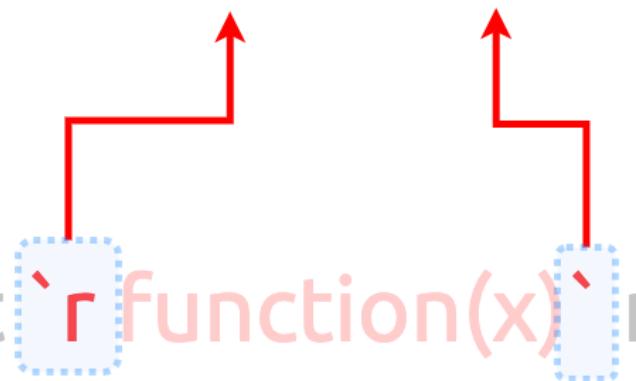
Previous **1** **2** **3** **4** Next

Inline R Code



R Markdown also supports inline R code

Grave accent or back-tick
around code



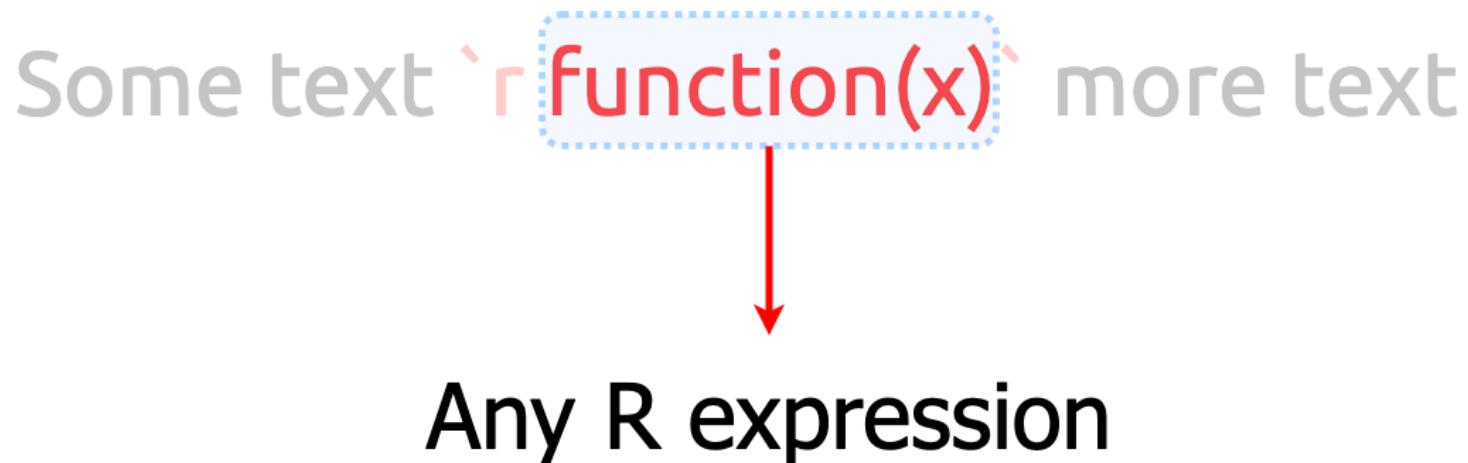
The diagram illustrates the syntax for inline R code. A sentence "Some text" is shown in gray. In the middle, there is a pink word "function(x)" enclosed in blue dashed brackets. To the left of this bracketed code, there is a red grave accent character (`) enclosed in a blue dashed bracket. To the right of the bracketed code, there is a red back-tick character (`) enclosed in a blue dashed bracket. Red arrows point from each of these three red characters up towards the pink "function(x)" word, indicating they are all part of the same inline R code block.

Some text `r function(x)` more text

Inline R Code



Inline R code allows us to include summaries of our analysis in the report



Exercise 9: Add Inline Code



We're going to add a Pearson correlation between speed and stopping distance to the **01-monthy-report.Rmd**

Include the following code under the **Summary** level three header

```
The correlation between speed and stopping distance  
is `r cor(x = cars$speed, y = cars$dist)`
```

Knit the document again

Exercise 9: Add Inline Code (rendered)



Monthly Report

Martin Frigaard
10/27/2020

R Markdown

This is a **monthly report** generated with **RMarkdown**, a literate programming tool for combining text and `code`.

Summary Tables Including Plots

The correlation between speed and stopping distance is 0.8068949

```
## Summary
```

```
The correlation between speed and stopping distance
is `r cor(x = cars$speed, y = cars$dist)`
```

```
```{r cars}
summary(cars)
```

```

```
summary(cars)
```

```
##      speed          dist
## Min.   :4.0   Min.   : 2.00
## 1st Qu.:12.0  1st Qu.:26.00
## Median :15.0  Median :36.00
## Mean   :15.4  Mean   :42.98
## 3rd Qu.:19.0  3rd Qu.:56.00
## Max.   :25.0  Max.   :120.00
```

Make cool stuff in R Markdown!

bookdown

blogdown

these slides!



Resources



- **YAML**: check out the [yamlthis package](#) for tools and documentation for working with YAML
- **Markdown**: [Commonmark](#) has a quick ten-twenty minute tutorial on markdown.
- **R Markdown**: A comprehensive but friendly introduction to R Markdown and friends. Free online!
- **R for Data Science**: A comprehensive but friendly introduction to the tidyverse. Free online.
- **R Markdown for Scientists**: R Markdown for Scientists workshop material